

3D Support for Business Process Simulation

Daniel Eichhorn, Agnes Koschmider, Yu Li, Peter Stürzel,
Andreas Oberweis, Ralf Trunko

Institute of Applied Informatics and Formal Description Methods
Universität Karlsruhe (TH)

76128 Karlsruhe, Germany

{eichhorn|koschmider|li|stuerzel|oberweis|trunko}@aifb.uni-karlsruhe.de

Abstract—Simulation is a key technique for the design and redesign of business processes and is used to evaluate new business process models with respect to key performance indicators. The increasing complexity of today’s business processes hampers a quick identification of weak points of a business process design that were uncovered during simulation. In this paper we add a third dimension into the description of process and data objects, which enables a compact 3D view of objects in process modeling as well as simulation. The benefits are statistical analysis of simulation results based on volume and size changes of 3D process and data objects as well as customized 3D simulation views. First evaluations indicate the effectiveness of our approach, which goes beyond conventional simulation tools for business processes.

I. INTRODUCTION

In the area of business process management, simulation is a key technique for design and redesign of business processes. Simulation is used to evaluate new business process models with respect to key performance indicators or to test process changes by performing what-if scenarios. In this way, simulation is a way to test decisions prior to their implementation in real business environment. Furthermore, simulation allows for the integration of variability and uncertainty into the anticipation of business process performance [1]. A simulation experiment is based on the introduction of dynamic process parameters (e.g. durations, capacities, volumes, work rates, budgets) whose changes are gathered by using specific process metrics. Thus, business process simulation bridges the gap between assessment of existing and design of projected process models [2] and allows for the measurement of process performance. Process performance indicators are e.g. throughput, costs, inventories, cycle times, resource and capital utilization, start-up times, cash flow, or waste [1].

Simulation capabilities are offered by a growing number of business process management tools to enhance their analytical functions [3]. Such tools provide users with a variety of analysis possibilities for simulation runs based on standard process performance metrics. But the increasing complexity of today’s business processes hampers the quick visual allocation of weak points in a business process

model, especially for inexperienced users. User-friendliness as well as modeling, simulation and output analysis capabilities are seen (amongst others) as key criteria for the usability of business process management tools [4].

In our approach, we aim at a compact visualization of business process simulation and its results by adding a third dimension into the description of data and process objects and linking classical key performance indicators for business processes with the concept of 3D process visualization. In this context, we enhance our concept for spatial visualization of Petri net diagrams, where we extended the conventional “flat” representation of business processes modeled with Petri nets with a third modeling dimension [5]. Petri nets are a well-founded process modeling language with formal semantics and a graphical representation. A Petri net is a directed bipartite graph in which nodes represent places and transitions, and tokens may be assigned to places.

Our approach enables interactive 3D animations of business process models based on Petri nets as well as statistical analyses of simulation results based on volume changes of 3D process and data objects. In this way, we support users to quickly identify weak points of modeled business processes. We decided for a 3D visualization of business processes since it supports the human visual intuition and allows for integrating more information in a single business process model [19]. In addition, it is possible to offer different views of a business process model by executing simple geometric operations and crossing of arcs can be avoided [26]. In the context of business process simulation, visual information can impart complex contents easier than packages of texts and numbers. A 3D visualization of simulation results is striking and intuitive for the users. In this context, our approach supports an effective graphical representation of the results of simulation runs, which allows an easier interpretation of these visualized information.

The remainder of this paper is structured as follows. Section 2 describes the 3D representation of process and data objects used for performance measurement. Within Section 3, we describe performance measurement metrics for data and process objects based on a 3D visualization of business processes. In Section 4, the analysis possibilities of 3D simulation results are discussed. In Section 5, we survey

related work in the field of 3D visualization of business process models, business process simulation techniques, and metrics for performance measurement of business processes. The paper concludes with a summary and gives an outlook on future research.

II. 3D REPRESENTATION OF PROCESS AND DATA OBJECTS

In [5], an approach has been presented that introduces a third dimension into business process models in order to represent the relationship between a process and an organizational model or to display semantic relationships between two business process models. In this paper we will introduce a third dimension into the description of process and data objects in order to support a compact 3D modeling and simulation of business processes.

Generally, in the context of business process management, objects in business processes can be classified into data objects and process objects [6]. *Data objects* refer to flowing objects conveying data that are manipulated and delivered across a process net. The delivery of data objects (possibly with data transformation) forms the data flow of business processes. In Petri nets, data objects are often modeled as (attributed) tokens that move through the process net by enabling transitions. *Process objects* are non-flowing objects used to construct the control flow or serving as parameterized indicators for restricting, analyzing, executing, monitoring or improving business process models. Process objects in (high-level) Petri nets consist of net elements (i.e., places, transitions and arcs) and process constraints or performance indicators such as cost, time, probability, roles, resources, arc weights or place capacities.

In this Section we first define process and data objects and then individually discuss in detail process objects and their representation in 3D environments. The reason why we choose cost, time, resources and place capacity for discussion is that from our viewpoint they are fundamental key performance indicators for monitoring and analysis of business process models.

In our approach data objects (tokens in the Petri net) are represented as realistic 3D objects that are automatically generated according to their attributes specified by the modeler. In many process modeling and simulation scenarios, this is helpful for the user to observe and validate transformation processes of data objects in virtual reality. For instance, in the simulation of product assembly process models, one can zoom into places and examine data objects from components to completed products from different perspectives. The assembly process can thus be easily understood, and possible assembly errors or flaws can be perceived intuitively.

Process objects are represented as geometric 3D figures, e.g. places as spheres, transitions as cubes. In this paper we define the process objects assigned to dynamic process components (in our context transitions) as a tuple $PD = \langle T, f_{cost}, f_{time}, f_{resource} \rangle$ where

- T is a finite set of transitions;
- f_{cost} is a cost function $f_{cost}: T \times \mathcal{I} \rightarrow \mathbb{R}_+$ where \mathbb{R}_+ is the set of nonnegative real numbers and \mathcal{I} is a finite set of indices of transition occurrences;
- f_{time} is a time function $f_{time}: T \times \mathcal{I} \rightarrow \mathbb{R}_+$
- $f_{resource}$ is a resource function $f_{resource}: T \rightarrow P(R)$ with R a finite set of resources and $P(R)$ the power set of R .

The process objects assigned to static process components (in our context places) are defined as a tuple $PS = \langle P, f_{capacity} \rangle$ where

- P is a finite set of places;
- $f_{capacity}$ is a capacity function $f_{capacity}: P \rightarrow \mathbb{N} \cup \{\infty\}$.

A. Cost

Transition cost C_{trans} is a function as defined above and consists of a fixed and a variable cost component, i.e.,

$C_{trans}(t, i) = C_{fix}(t) + C_{var}(t, i)$ with $t \in T, i \in \mathcal{I}$. Fixed cost is the constant expense of a transition that does not change in simulation. In contrast, variable cost varies during simulation and is calculated according to continuous uniform distribution on an interval $[a, b]$ with $a, b \in \mathbb{R}_+$. Fixed cost and interval boundaries of variable cost are assigned by the modeler prior to simulation.

Figure 1 shows a 3D representation of cost factors in a simulation run. To adequately take advantage of 3D visualization we represent each cost factor as a cylinder. The height of the cylinder varies according to current values of its corresponding cost indicators. The cost cylinder is included in a transparent cylinder that controls the increase/decrease of cost factors (e.g., if a cost factor exceeds the maximum costs assigned by the user, then the cylinder will change its color and can only expand within the boundaries of the transparent cylinder). The change of color of geometric figures will be explained in Section 3.

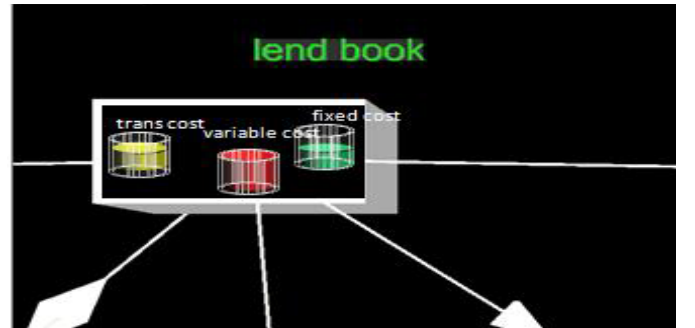


Figure 1. 3D representation of costs

B. Time

Transition time refers in our approach to the time interval from removing tokens from pre-set places to

creating tokens in post-set places. It is a function which consists of three components, i.e.,

- $T_{\text{trans}}(t, i) = T_{\text{pre}}(t, i) + T_{\text{dur}}(t, i) + T_{\text{post}}(t, i)$ with $t \in T, i \in J$ and
- T_{pre} representing transition occurrence preparation that can be interpreted as the time needed by the transition to wait for necessary data and resources;
 - T_{dur} describing the duration of a transition occurrence;
 - T_{post} referring to the time needed by the transition to release used data and resources.

Transition time can be either deterministic or stochastic (i.e., described by some probability distribution function). In our approach, transition time (including its three parts) is stochastic and calculated by using the exponential random distribution supported on three intervals for $T_{\text{pre}}, T_{\text{dur}}$ and T_{post} , respectively. Boundaries of the intervals are nonnegative real numbers and given prior to simulation.

To represent transition time in 3D environment, we display over each transition a 3D area diagram for $T_{\text{trans}}, T_{\text{pre}}, T_{\text{dur}}$ and T_{post} as shown in Figure 2. The y-axis shows the elapsed time, the x-axis the indices of transition occurrences, and the z-axis the types of transition time indicators. The user can rotate the area diagrams for different view perspectives.

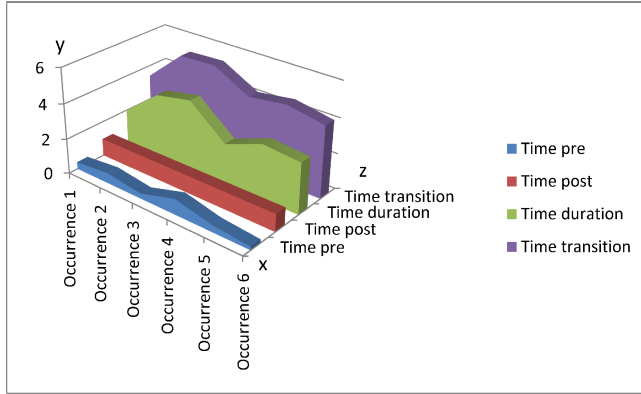


Figure 2. 3D representation of transition time

C. Resource

Tasks defined in transitions are carried out by human resources or using nonhuman resources (e.g., machinery, instrument, computer, software system etc.). A resource can be assigned to for more than one transition. The *resource availability* is defined in our approach as a function $A: R \rightarrow I$ where $I = \{[t_i, t_j] \mid t_i \in \mathbb{R}_+, i, j \in \mathbb{N}, t_i < t_j \text{ if } i < j\}$ is a set of non-overlapping time intervals. The availability of a resource for a transition can be defined in a similar way, i.e., $A_{\text{trans}}: R \times T \rightarrow I$.

The total *available time* of a resource for a process is $AT: R \rightarrow \mathbb{R}_+$ and can be calculated by summing up the time intervals (i.e., $t_j - t_i$) of A . Similarly, the available time

for a transition is defined as $AT_{\text{trans}}: R \times T \rightarrow \mathbb{R}_+$. The *load* of a resource for a transition is a function $L_{\text{trans}}: R \times T \rightarrow \mathbb{R}_+$ that can be calculated with $L_{\text{trans}}(r, t) = RT_{\text{trans}}(r, t) / AT_{\text{trans}}(r, t)$ where $RT_{\text{trans}}: R \times T \rightarrow \mathbb{R}_+$ is the elapsed time used by a resource for a transition.

To show resources with their time attributes related to transitions in 3D environment, we display over each transition a sequence of overlapping icons representing human or non-human agents, i.e., $icon_i$ with $i \in J$ the index of transition occurrences (see Figure 3). The icons are two-dimensional and same-sized. The size is proportional to the value of AT_{trans} and remains constant in a simulation. Each icon is filled with colors such as green, yellow or red for warning purpose. The filling level varies according to current load of the resource. Different filling color is displayed according to user-defined ranges. For example, if the current load is under 70%, the icon for current transition occurrence is filled with green. If it is between 70% and 95%, yellow color is filled. The icon is filled with red if the load is over 95%. An alert will be triggered in simulation if the icon brims over with color, i.e., the current load is above 100%. Then a warning text highlighted in red will occur in the transition.

The icon for the latest occurrence is always displayed in the front of an icon stack. To acquire resource load information of previous occurrences, the user can scroll forward on the figure stack to take a closer look at corresponding figures.

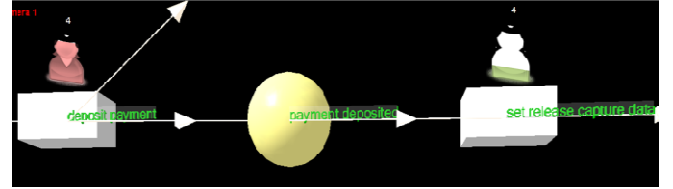


Figure 3. Assignment of Resources to Activities

D. Capacity

Place capacity restricts the number of tokens that are allowed to be contained in a place. In our approach, it is deterministic and should be assigned by the modeler. If not given, it is infinite by default as defined before.

We represent places as 3D balls whose size is determined by their finite capacity, i.e., the length of the sphere radius depends on the value of the capacity. If the capacity is infinite, then the place is displayed as non-transparent sphere, i.e., tokens in the place are invisible. Transparent places are filled with tokens that are displayed as small balls, or realistic 3D objects if related attributes are defined. The tokens are piled from the bottom of the place (see Figure 4), which allows the user to observe the filling level that shows current rate of capacity utilization. To better alert possible capacity bottlenecks, we color tokens with green, yellow and red indicating whether there is much,

little or no room left in capacity, respectively. The ranges indicated by these 3 colors are defined by the user.

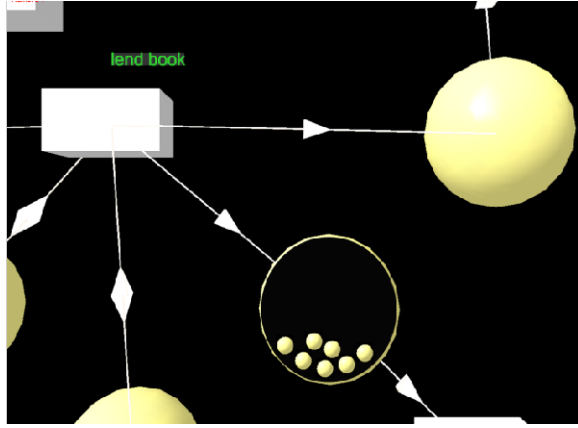


Figure 4. 3D Representation of Capacity

III. FORMING DATA AND PROCESS OBJECTS IN 3D SIMULATION ENVIRONMENTS

3D simulation supports quick identification of weak points of a business process design. However, this identification and moreover the understandability of 3D simulation results may be hampered if no additional mechanisms are provided, which control the increase or decrease of the size or volume of geometric figures during the simulation.

In this section we will present our approach of a controlled visualization of changes of geometric figures representing process and data objects. Additionally, metrics will be presented that support an individual adjustment, visualization of simulation data and comprehensiveness of 3D simulation results.

A. Size and Volume of 3D Data and Process Objects

Process and data objects are modeled in our scenario as geometric figures or as a 3D area diagram. Costs are represented as cylinders, time by a 3D area diagram, resources by icons (made up of several geometric figures such as cylinders and ellipsoids) and capacities as spheres. To visualize weak points of the process design through simulation we will change the volume v or the size s of these figures. In our scenario the expansion of s or v area indicates an increase of an object parameter; e.g., if costs increase then the volume of the cylinder will expand.

To enable users to visualize changes of process and data objects we vary the volume of cylinders (referring to costs), and of resources and the size of area diagram (referring to time) and of a sphere (representing capacities).

B. Monitoring of 3D Data and Process Objects

Based on the volume v and the size s of these figures (respectively the diagram) we define a formula that changes

s or v of the figures in simulation. The changes of figures are performed automatically by a tool (depending on the change of costs, time, capacities and resources in the simulation). Initially, all parameters p controlling the size and the volume (p_h, p_l, p_r, p_w) have values in $[0..1]$ range, with their default setting 0.5. Hence, each figure has a default size and volume computed from its corresponding default ps . The modification for each p is defined by:

$$\text{modification } p = \frac{c * \Delta \text{ objectUnit}}{\text{objectUnit}}$$

where *objectUnit* represents the total sum of costs, time, resources or capacity. $\Delta \text{ objectUnit}$ is the relative modification of costs, time, resources or capacities; $c \geq 0$ and the value of c correlates negatively with the height of an *objectUnit*.

To monitor the current status of an *objectUnit* in a simulation environment we use three colors for the size or volume of the geometric figures or diagrams:

- Green: the value is performing well,
- Yellow: warning that a value indicates a critical degree,
- Red: alarming that a value indicates an impact problem.

To react to critical situations or problems, which may occur, the user needs to specify the values that turn a light from green to yellow and then to red. Usually, before starting a simulation the user assigns values e.g., C_{\min} for activity “produce order” is 10 € and C_{\max} is 20 €. This cost indicators are regarded as the boundaries for C_{trans} and are responsible for changing the color of the (cost) volume cylinder. Based on the monitoring results we can measure the violation of interval boundaries for a process or data object by:

$$\frac{\text{fraction of time in yellow area}}{\text{fraction of time in green and yellow area}}$$

where a high ratio recommends that an action (i.e. exception handling) should be initiated.

The monitoring of process and data objects can also be obtained by calculating the volume difference of geometric figures.

C. Metrics for 3D Data and Process Objects

This subsection proposes metrics for 3D data and process objects in simulation environments. These metrics affect the visualization, adjustment of simulation data and the comprehensiveness of 3D simulation results.

1) Prioritization Number

Each user may have a different perception regarding simulation criteria. Therefore, users can rank objects to be

displayed in the foreground. This formula is used if simulation analysts prefer observing some objects more than others and not in the sense to avoid overlapping of objects. Even if the user will change the perspective (moving the view in the 3D environment) the prioritized objects will be more highlighted than the less prioritized objects. Subsequently, a list of objects in descending order can be defined by calculated priorities. The objects at the top of the list have the most favorable balance of value, or cost and thus - all other factors being equal - should have highest priority.

The priority metric is used during the simulation to customize the simulation analyst's view on the simulation run.

2) Control Flow Complexity Metric

Business processes can be complex to simulate (number of control flows or nesting depth of control flow structures). This metric computes the complexity degree required to simulate a process. Generally, the degree depends on the amount of data and process objects, which affects the control flow, assigned by the user. A very popular technique to measure the number of all possible control flows is the *cyclomatic number* introduced by McCabe [7]. This number measures the number of linearly independent execution paths through a software program. For a well-structured software module, the cyclomatic number is simply one plus the number of branching statements (binary decisions, e.g., IF-statements in a programming language) in the module.

3) Role/Resource Metric

Usually, process modeling requires defining an organizational model that comprises the relationship between roles and resources (which belong to an organizational unit). Roles are then assigned to process activities in order to connect the process and organizational model. A Role Metric computes the degree between the fraction of assigned roles in the business process model belonging to an organization unit and the fraction of all roles belonging to an organizational unit:

$$\frac{\text{fraction of assigned roles}_{OE}}{\text{fraction of all roles}_{OE}}$$

The Resource Metric computes the degree between assigned resource and all available resources:

$$\frac{\text{fraction of assigned resources}}{\text{fraction of all resources}}$$

Based on these ratios we can determine the disposal of roles or respectively resources in a business process. Especially our representation of data and process objects in 3D allows highlighting roles belonging to the same organizational units and thus facilitates shifting of roles as Figure 5 shows. The assignment of a process model to an organizational model in 3D has been described in [5].

As an example, in Figure 5 two activities are performed by roles of the same organizational unit. One role is overloaded, which is shown by a red colored figure. The other role is capable of executing more tasks (represented by a green color and the volume filling). Both roles are assigned to a high-level role, which is performing well (represented by a green color). Consequently, the user can shift tasks from the red colored role to the green colored role.

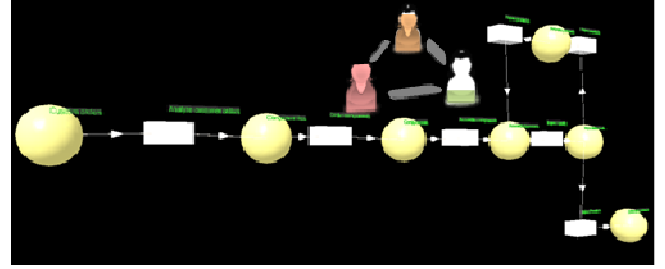


Figure 5. Highlighting of relationships between roles

The role/resource metric is applied during the simulation run but can also be used when the user is observing the analysis results (as explained in the next section).

4) Data Flow Complexity

To calculate the data flow complexity we use the information flow complexity of Henry and Kafura [8]:

$$\text{Information flow complexity}(M) = \text{length}(M) \times (\text{fan-in}(M) \times \text{fan-out}(M))^2$$

where transferred to our application area fan-in can be regarded as the number of all modules that call a given module and fan-out is the number of all modules that are called from the process model to be simulated.

The fan-in of a business process M is the number of data flows that terminate in M, plus the number of places from which information is retrieved by M. Similarly, the fan-out of M is the number of data flows that emanate from M, plus the number of places from which data are updated by M.

The control flow and the data flow complexity degrees for a specific business process can be computed before starting the simulation and can then be displayed during the simulation or when the user is observing the analysis results (see next section). For the second application case a high control flow complexity degree may demand the user to improve his process model.

5) Cognitive Complexity Metric

Cognitive complexity is situation-specific and affects the user's individual perception. Individuals who are described being cognitively complex in interpersonal domains are generally hypothesized to have stronger (social) perception skills [9]. [10] defines cognitive weight as a complexity metric to measure the difficulty or relative time and effort required for comprehending a given piece of software modeled by a number of basic control structures.

In a 3D simulation environment cognitive weights can be regarded as a degree that measures the effort required to comprehend a simulation result. Especially the user needs to comprehend the meaning of the colors, the surface or the volume and the caption of the three axes. Then the cognitive weight of simulation results is defined as the sum of the cognitive weights for these three comprehension factors.

IV. ANALYSIS OF SIMULATION RESULTS

In Section 2 we described our approach of a 3D representation of process and data objects used for performance measurement. In Section 3 we presented our approach of a controlled visualization of changes of process and data objects during performance measurement. Additionally, metrics were introduced that support an individual visualization of simulation data and the comprehension of simulation results.

In this Section we will describe how to combine the process and data objects described in Section 2 (costs, time, capacity and resources) with the metrics presented in Section 3 in order to obtain a compact 3D analysis of simulation results.

A. Analysis and Monitoring

The aim of a 3D representation of analysis results is a quicker understanding of the simulation data set. Especially large amounts of information could be displayed easily understandable. In 3D there are more possibilities to present information than in 2D. For example, our time cylinder diagram can be easily drawn in another cylinder diagram at a special place, which should be emphasized. This possibility also exists in a 2D cylinder diagram, but since we represent on the base of three axis (x,y,z) we could not locate the exact position in 2D. Another benefit of our 3D representation of simulation results is that we are able to integrate all simulation views into one 3D diagram. Current tool implementations are using several different tabs to display different simulation results.

For the 3D analysis of simulation results we exploit the x, y and z-axis for visualizing different objects. E.g., in Figure 6 the x-axis is used for resource allocation (in percent), the y-axis for roles and the z-axis for the height (the intention of the resource allocation).

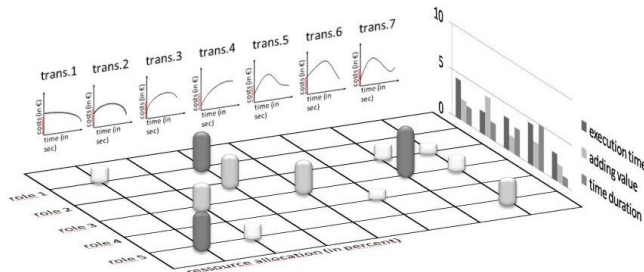


Figure 6. 3D Analysis of Role Performances

The white, grey and black geometric figures in the middle diagram reflect the degree of allocation of roles for specific activities respectively transitions (note that the colors of the geometric figures can be replaced by green (instead of white), yellow (grey) and red (instead of black) colors). The diagrams displayed under the headlines (*trans.n*) show the degree between time required to perform a task and the costs for the roles assigned to this activity. The right hand side of Figure 6 displays a diagram with values like execution time, adding value and time duration for each role.

Figure 7 shows another example of 3D analysis of simulation results focusing on activities. Like Figure 6, the white, grey and black geometric figures in the middle reflect the activity performance in the process simulation. The diagrams with headlines (*trans.n*) are simultaneous to Figure 6 (same process simulation). The diagram on the right side represents the data flow complexity and the control flow complexity calculated according to different activity performances.

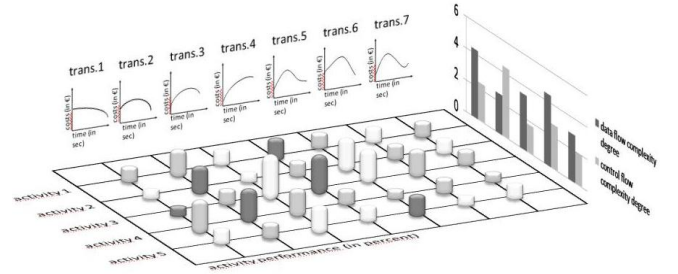


Figure 7. 3D Analysis of Activity Performances

B. Customization of Analysis Results

A user should have the possibility to change the representation of the 3D analysis. It should be possible to generate an easy to understand view and if required the individual usage of 3D objects (which might decrease the Cognitive Complexity Metric degree). One possibility for customizing analysis results can be obtained by a scroll-in function of a diagram. For example in Figure 6, the user should be able to choose the right diagram and displaying it in the middle for a better recognition of the details. The different diagrams should be implemented as standalone widgets. These widgets could be used in different view modes.

V. RELATED WORK

Existing work relevant for our approach can be differentiated into three categories: (1) Process simulation (regarding cost, time, resources, and capacities), (2) 3D representation and (3) Performance analysis.

Regarding the simulation of models in general, the set of approaches found in the literature is numerous [3, 11, 12, 13]. Proposals for building up a simulation model for business processes can be found in [3]. More specifically,

[12] proposes a concurrent algorithm to obtain approximate results with higher simulating performance and less errors. [11] suggests to describe the simulation model and the language-action model, which is a model including resources like actors and objects and its coordination between. However, the simulation tools do not provide a representation and a simulation of a three-dimensional business process model with formal semantics.

Three-dimensional analysis has been suggested for data-models [14, 15]. For instance, graphical representation techniques for output data of data models can be found in [14], which is limited to handle complexity and to obtain good results quickly. [16] describes a three-dimensional representation of data models with data sets in general. Thus, these approaches cannot be utilized in our scenario.

The technique of separating different parts of a given model in different views is a common technique to handle the complexity of business process models [17, 18]. Several tools have been proposed which use the third dimension for business process modeling [19, 20, 21]. They support the integration of new objects and also a message transfer for user interactivity. An approach for a 3D compact representation of dependencies between objects, organizations and process models can be found in [5].

For literature on performance measurement or performance analysis we refer to [22, 23, 24]. In [22] a business process cockpit on top of a data-warehouse system is presented that helps users analyzing business data properly. [22] is focusing on the business process specific monitoring and analysis, but in contrast to our approach with a concrete data warehouse system on the basis. We are focusing on a Petri Net in general. [23] describes a similar paper with a focus on collaborative work (different roles) and a multi-dimensional visualization. But the work of [23] is limited to role consideration in contrast to our paper that considers more objects in the 3D simulation and analysis representation. [24] explains how process-oriented business performance management could be modeled, analyzed and monitored with higher Petri nets.

VI. CONCLUSION

In this paper we added a third dimension into the graphical representation of process and data objects, which enables a compact 3D view of objects in process modeling as well as simulation. The benefits of this are statistical analysis of simulation results based on volume and size changes of 3D process and data objects as well as customized 3D simulation views.

Within these scenarios we showed that the 3D representation of processes facilitates the access to process-specific information. Whilst in conventional 2D representation process-specific information interfere with each other, in 3D the information becomes straightforward. By turning the process model in its 3D environment, one can examine different views and gather easily process-specific information.

Future work comprises the evaluation of our approach and the integration of the implemented prototype into the

Petri net-based process modeling framework INCOME2010 [25]. Evaluation comprises the execution of simulation runs of different process models, analysis of the results as well as discussion of our approach with selected test users. In addition, 3D visualization and animation of other process objects (e.g. process metrics such as time, cost, etc.) will be explored in our future research. In this contribution, we are focusing on a 3D representation concerning the control and data flow of processes. A further step in the future will be a 3D representation concerning especially the data flow of processes (e.g. XML documents in high level Petri nets).

ACKNOWLEDGMENT

The second author would like to thank the Karlsruhe House of Young Scientists that has partially financed the research reported in this paper.

REFERENCES

- [1] April, B.; Better, M.; Glover, F.; Kelly, J.; Laguna, M.: Enhancing Business Process Management With Simulation Optimization. In: Proceedings of the 38th Conference on Winter Simulation, pp. 642-649, 2006.
- [2] Farrington, P.; Nembhard, H.; Sturrock, D.; Evans, G.: Business Process Simulation: A Fundamental Step Supporting Process Centred Management. In: Proceedings of the 1999 Winter Simulation Conference, Phoenix, Arizona, United States, pp. 1383-1392, ACM, 1999.
- [3] Jansen-Vullers, M.; Netjes, M.: Business Process Simulation – A Tool Survey. In: Proceedings of the Seventh Workshop and Tutorial on the Practical Use of Coloured Petri Nets and the CPN Tools, Volume 579 of DAIMI, pp. 77-96, 2006.
- [4] Bradley, P.; Browne, J.; Jackson, S.; Jagdev, H.: Business Process Reengineering (BPR) – A study of the software tools currently available. In: Computers in Industry, 25(3), pp. 309-330, 1995.
- [5] Betz, S.; Eichhorn, D.; Hickl, S.; Klink, S.; Koschmider, A.; Li, Y.; Oberweis, A.; Trunko, R.: 3D Representation of Business Process Models. In: Proceedings of Modellierung betrieblicher Informationssysteme, LNI P-141, Gesellschaft für Informatik, Bonn, pp. 73-87, 2008.
- [6] van der Aalst, W. M. P. and Berens, P. J. S.: Beyond workflow management: product-driven case handling. In: International ACM SIGGROUP Conference on Supporting Group Work, ACM Press, New York, pp. 42-51, 2001.
- [7] McCabe, T.J.: A complexity measure. IEEE Trans. Software Engineering, vol. 2, pp. 308-320, 1976.
- [8] Henry, S., Kafura, K.: Software structure metrics based on information flow. IEEE Transactions on Software Engineering, 7(5), pp. 510-518, 1981.
- [9] Burleson, B.R., Caplan, S.E.: Cognitive complexity. In Communication and personality: Trait perspectives. Cresskill, NJ: Hampton Press, pp. 233-286, 1998.
- [10] Shao, J., Wang, Y.: A new measure of software complexity based on cognitive weights. IEEE Canadian Journal of Electrical and Computer Engineering, 28(2), pp. 69-74, 2003.
- [11] Rittgen, P.: Deriving Simulation Models from Business Process Models. In: INFOCOMP Journal, 4 (3), pp. 23-31, 2005.
- [12] Horvath, L.; Rudas I.J.: Evaluation of Petri Net Process Model Representation as a Tool of Virtual Manufacturing. In: Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics, Information, Intelligence and

- Systems, IEEE, San Diego, USA, Volume 1, pp.178-183, 1998.
- [13] Yang, J.; Fujimoto, Y.: Modeling, Implementation and simulation of virtual factory based on colored timed Petri net. In: Emerging Technologies and Factory Automation, Proceedings ETFA'03, IEEE Conference, Volume 1, pp. 574-579, 2003.
 - [14] Grier, D.A.: Graphical techniques for output analysis. In: Proceedings of the 24th conference on Winter simulation, Arlington, Virginia, United States, ACM, pp. 314-319, 1992.
 - [15] Chambers, J.M.; Cleveland, W.S.; Kliener, B.; Tukey, P.A.: Graphical Methods for Data Analysis, Wadsworth International, Dusbury Press, 1983.
 - [16] Borgelt, C.; Kruse, R.: Graphical Models: Methods for Data Analysis and Mining. John Wiley and Sons, 2002.
 - [17] Scheer, A.-W.: ARIS – Business Process Modeling, Springer Verlag, Berlin, 1999.
 - [18] Bobrik, R.; Reichert, M.; Bauer, T.: View-based process visualization. In: 5th International Conference on Business Process Management, LNCS 4714, Springer Verlag, pp. 88-95, 2007.
 - [19] Ballegooij, A.; Elliens, A.; Schönhage, B.: 3D Gadgets for Business Process Visualization – A Case Study. In: Proceedings of the fifth symposium on Virtual reality modelling language, pp. 131-138, 2000.
 - [20] Krallmann, H.; Gu, F.; Mitritz, A.: ProVision3D - Eine Virtual Reality Workbench zur Modellierung, Kontrolle und Steuerung von Geschäftsprozessen im virtuellen Raum. Wirtschaftsinformatik, 41(1), pp. 48-57, 1999.
 - [21] Kindler, E.; Páles, C.: 3D-Visualization of Petri Net Models: Concept and Realization. In: Proceedings of International Conference of Applications and Theory of Petri Nets, Springer, pp. 464-473, 2004.
 - [22] Sayal, M.; Casati, F.; Dayal, U.; Shan, M.-C.: Business Process Cockpit. In: Proceedings of 28th International Conference on Very Large Data Bases, Hong Kong, China, VLDB Endowment, pp. 880-883, 2002.
 - [23] Yin, Y.; Qin, S.; Holland, R.: Development of a project level performance measurement model for improving collaborative design team work. In: Proceedings of the 12th International Conference on CSCW in Design, IEEE, pp. 135-140, 2008.
 - [24] Mevius, M.; Oberweis, A.: A Petri-Net Based Approach to Performance Management of Collaborative Business Processes. In: 16th International Workshop on Database and Expert Systems Applications, IEEE, pp. 987-991, 2005.
 - [25] Klink, S.; Li, Y.; Oberweis, A.: INCOME2010 - a Toolset for Developing Process-Oriented Information Systems Based on Petri Nets. In Proceedings of International Workshop on Petri Nets Tools and Applications. ACM digital library, Marseille, France, March 2008.
 - [26] Rölke, H.: 3-D Petri nets – Making Use of 3 Dimensions in Executable Petri Net Modelling. In: Petri Net Newsletter, vol. 72, pp. 3-9, April 2007.